

Bernd Mehlis, DL3JMM

(Amateur) radio operation on frequencies below 8300 Hz

- General
- Legal notice
- RX technology / software / antennas
- TX technology / software / antennas
- Send types
- Grabber

1. General

The frequency range below 8300 Hz offers fascinating experiences in radio connections that only a few amateurs have made so far. Overcoming challenges that were initially thought to be insurmountable gives a "Marconi feeling" of the early days in radio. In the meantime, amateurs have achieved connections between Europe and Asia and between Europe and North America.

- SNR / Propagation / Frequency / Phase stability

Operating at these low frequencies is complicated by higher background noise, interference from worldwide thunderstorm activity, man-made interference, and the extremely poor efficiency of transmitting antennas.

Wave building propagation occurs due to ground waves along the curvature of the Earth and reflections from the ionospheric D-layer, which occupies different heights depending on the time of day. (approx. 65 km by day - 90 km by night).

Attenuation is relatively low, mostly lower at night, but stronger atmospheric disturbances prevail at night.

Because of the problems mentioned above, special methods must be used to make the signals readable nevertheless. Essentially, this is done by temporal integration with maximum possible narrowing of the bandwidth. Therefore, only very small data rates are possible and a lot of patience is needed for the experiments. For this reason, it is imperative to keep the frequency and even the phase absolutely constant when sending and receiving the signals. This is now possible thanks to GPS and is usually realized by evaluating the GPS time signals and the 1PPS clock.

2. Legal notice

- Country-specific allocations
- Generally no regulation / no permission (license) necessary
- Anyone could send

In most countries, radio operation below 8.3 kHz is not regulated, as in Germany, the frequencies can be used without permission as long as no strong harmonics interfere with higher frequencies. Thus, anyone could transmit on these frequencies. To be on the safe side, however, one should inform oneself about the respective regulations in one's own country beforehand.

3. Receiver technology / software / antennas

Usually a receiving system for these frequencies consists of one or more antennas, the necessary amplifiers / impedance converters, a computer with a sound card and the corresponding evaluation programs.

3.1 Devices, computers

- Sound cards

If only one antenna is used, a stereo (2-channel) sound card is sufficient for first tests. If one wants to achieve a possible cardioid formation with the help of further antennas by software calculation, in order to fade out interferences by the directional effect or if one wants to achieve a direction-dependent (colored) display of the signals in the waterfall diagram, 4 inputs of the sound card are recommendable.

For the evaluation of messages whose transmission often takes several hours (e.g. EbNaut), it is necessary to record the audio streams on a hard disk. For this reason, the interface of the sound card must ensure that, as far as possible, no interruptions occur in the stream due to other programs or activities in the operating system of the computers.

The sound cards should have a good SNR and offer at least 16 bit resolution at 48 kHz sample rate.

- Windows PC

With a Windows PC (usually as a laptop) one can receive signals of these frequencies over a suitable sound card, represent them in the waterfall diagram and record them after a filtering, noise blanking and data reduction also on the hard disk. WAV files can be generated to be evaluated or decoded after the transmission times have elapsed. If a Linux PC is used for reception, its audio streams can be decoded by means of "vorbis" encoding of the Windows evaluation software, and displayed in the waterfall case, for example.

- Linux PC (Raspberry Pi)

Advanced receiving systems can be realized with a Linux PC, mostly a Raspberry Pi is used for this and thanks to "vlfrc-tools" by Paul Nicholson (G8LMD) very good and mature software modules are available.

- GPS stabilization

As already mentioned in the introduction, a worldwide synchronization of the signals and receivers / transmitters is essential for the stabilization of frequency and phase. A GPS module (mostly from ublox) delivers the exact time and a signal with one impulse/second (1PPS), whose rising edge marks the exact beginning of the second. This should reach a channel of the sound card via the audio input, so that the antenna signals and the reference signal have exactly the same latency. Please follow the instructions for signal shaping of the application programs.

In order for the software to be able to assign the pulse to the correct second, the time of the computer should deviate by less than half a second. For permanent adjustment of the clock, the NMEA signals from the GPS can be read in via a serial or USB interface. Alternatively, the time can be obtained from the Internet using NTP, or from DCF77. The time is always measured in UTC.

3.2 Reception software

- Software for Windows

Here, for most applications, the program "Spectrum Lab" by Wolfgang Büscher (DL4YHF), which is equipped with very many functions, is used.

Programs: <https://www.qsl.net/dl4yhf/>

Description: <https://www.qsl.net/dl4yhf/speclab/index.htm>

- Software for Linux

Here the excellent tools of Paul Nicholson (G8LMD) are used: Program and

description: <http://abelian.org/vlfrx-tools/>

- Connection / transmission between computers

A connection can be established between the computers (Linux => Linux), (Linux => Windows) via WLAN or LAN (Ethernet) to transmit digital audio streams.

In addition, live audio streams from some external receiving stations can be fetched into the evaluation software.

- Linux => Linux: with the vt-tool "*vtcat*".
- Linux => Windows: with the vt-tool "*vtvorbis*".
- External stations: <http://abelian.org/vlf/index.php?page=access>

3.3 Receiving antennas

- Reception locations / faults

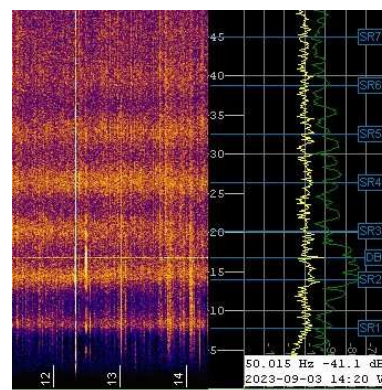
Because of the proximity to the conducting earth, only the vertical E-field component and the two horizontal H-field components are generally usable at very low frequencies.

Unfortunately, just this frequency range is very much affected by disturbances caused by the activity and installations of people. This starts with the supply of electric power, its frequency of 50/60 Hz and its numerous and strong harmonics, which cover the whole country in cables or overhead lines. Connected devices, photovoltaic systems with inverters, switching power supplies, etc. cause an enormous noise carpet, especially in urban areas.

Unfortunately, interference caused by thunderstorm lightning is unavoidable. Due to the high energy emitted by lightning and the low path attenuation, the interference signals extend over long distances. However, worldwide lightning discharges also contribute to the generation of Schumann resonances, whose **characteristic spectrogram** is suitable for testing the sensitivity of the receiving system at very low frequencies.

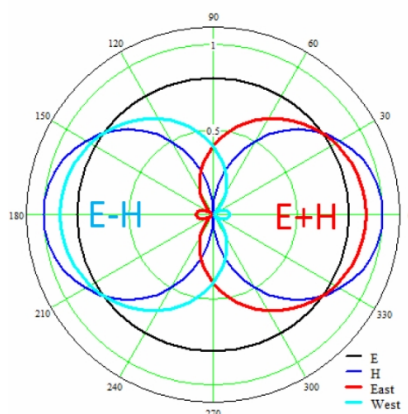
Only at a location with a very low level of interference and with a correspondingly sensitive receiving system can the Schumann resonances be received in this quality as in the spectrogram below.

For this reason, an attempt must be made to select the location for the antennas as far away as possible from any sources of interference.



- Optimization / interference suppression by cardioid formation

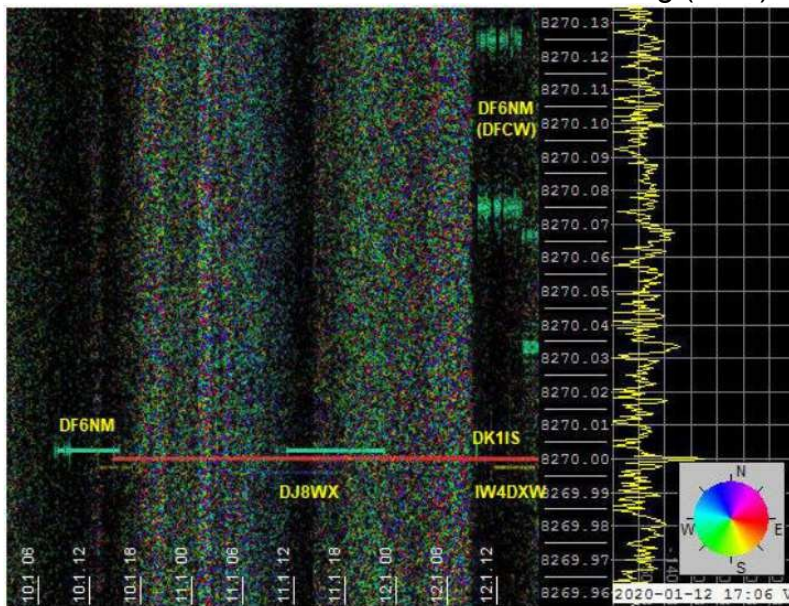
If 2 H-field antennas are set up orthogonally in (E/W) + (N/S) direction and a E-field antenna installed in addition, the signals can be added vectorially in the software. This allows a cardioid to be formed for suppression of rear QRN.



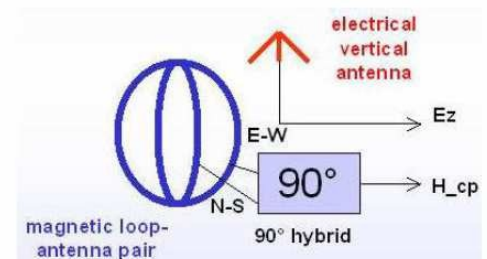
<http://abelian.org/vlfrx-tools/notes.html#Antenna%20synthesis>

In the waterfall diagram of "Spectrum Lab", direction-dependent colors for different signals can be displayed by means of these 3 antennas.

Antennas combined for Radio Direction Finding (RDF)



DF6NM (West), DJ8WX (Nord), DK1IS (Ost), IW4DXW (Süd)

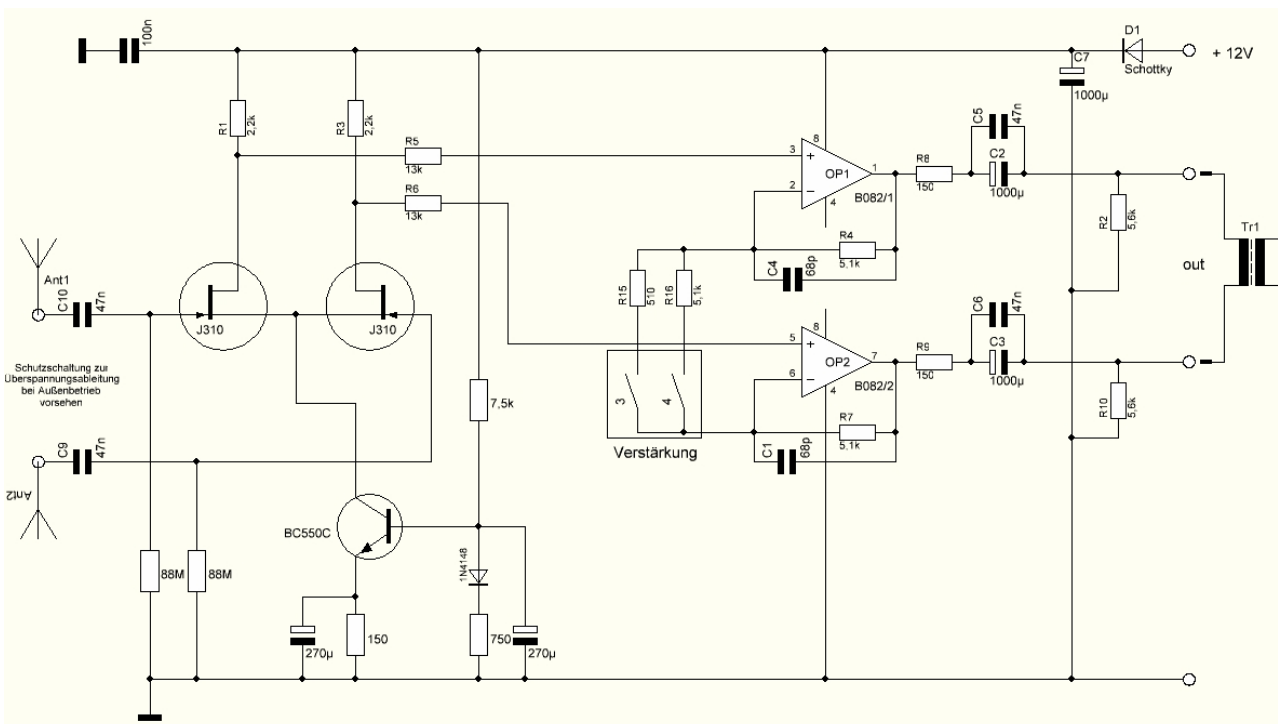


Farbpeiler von DF6NM:
Azimut => Phase => Farbe

90°-Kombination in Software

- E-Field antennas / impedance converter / amplifier

An E-field antenna consists of one or two capacitive receiving elements and a downstream impedance converter with very high input resistance. This can be followed by an amplifier. The signal from the antenna must be decoupled as well as possible from power supply signals during transmission to the sound card. For this reason, the antenna is often supplied with solar power from a battery and the signal is routed via isolating transformers. In my case, the signal is transmitted via an analog optical fiber link.



Symmetrical E-field probe:

2 tubes, each 1m long and 12mm in diameter, form the two vertical dipole arms. The inputs still have to be provided with protective circuits against lightning discharges.

Known forms of an E-field probe are the mini-whip, megawhip, etc.

Important Notice:

E-field antennas are to be set up as free-standing as possible! Taller trees or houses in the vicinity can significantly impair reception.

- H- field antennas / amplifiers

An H-field antenna always consists of an inductance and very different forms can be used: Ferrite antennas of different sizes, coils wound on MU metal up to more than 1m length, small and large loops made of several turns of wire, tube loops with 2m diameter, ground loops etc. are used as H-field antennas.

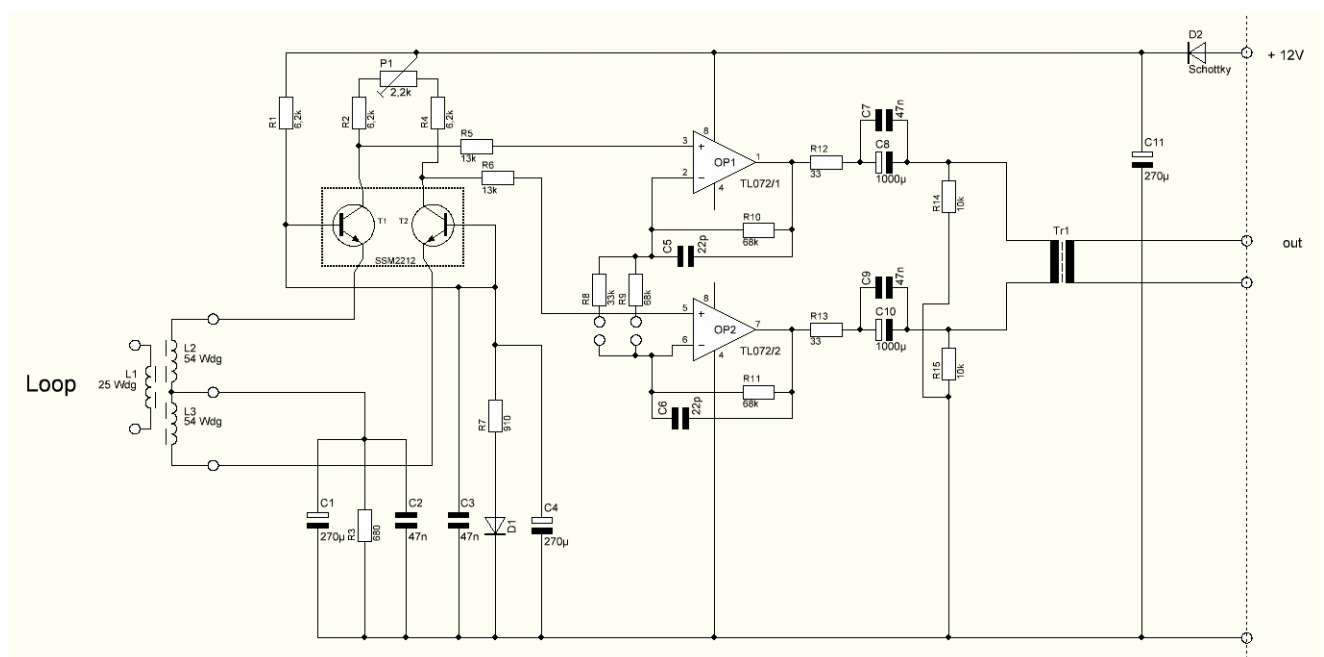


For me, a "triangle loop" with 8 turns of 2.5 mm² wire has proven itself. The leg length is 3 x 4m.

This is followed by an amplifier to compensate for transmission losses.

Here in my garden, 2 orthogonal triangular loops are hung in N/S and E/W directions in one place (on the tree).

The lower leg runs on the ground.



Amplifier for my triangle loop described above

Here, too, further protective circuits against overvoltages during thunderstorms must be provided.

Various suggestions for H-field antennas can be found here:

<http://www.vlf.it/>

4. Transmitting technology

For transmitting on frequencies < 8.3 kHz, audio power amplifiers are adapted to the respective antennas. This is usually done by transformation with iron transformers or, at higher frequencies, with ferrite transformers. In the case of Marconi antennas, resonance matching must be performed by loading coils, which can assume significant dimensions due to the required Q and loss ar- mut. For ground loops, a simple transformer with a capacitor in series with its secondary winding can provide matching to the loop. By driving with a clean sinusoidal signal, the formation of impermissible harmonics can be reduced. A distance of more than 1000 km has been bridged at 8270 Hz with 5 W already. But higher output power can be useful. The current world records on 8270 Hz and 5170 Hz were achieved with 300 W.

<http://abelian.org/vlf/amateur-radio/>

4.1 Devices, computers

- Sound cards

The control of an audio amplifier for broadcasting purposes can be generated by sound cards. For Windows the program "Spectrum Lab" offers possibilities of signal generation to enable various types of transmission. Do not forget to connect the PPS signal to the analog input.

In Linux, the sound card is controlled by the program "*ebsynth*" and outputs highly stable sinusoidal signals for carrier transmissions and for EbNaut. For both Windows and Linux, an analog input must be assigned with a PPS signal in order to generate highly stable and phase-locked signals.

When sending, only a sound card with 2 analog inputs is required. (stereo)

- Windows PC

Different programs can run here to control various types of transmission. Mostly "Spectrum Lab" which generates audio signals with the sound card or "*ebnaut-tx.exe*" which is used to control the phase switching needed for EbNaut via a RS232 interface.

http://abelian.org/ebnaut/software_ms.shtml

- Linux PC (Raspberry Pi)

There is for EbNaut and unmodulated carrier transmission the advantageous program "*ebsynth*", again developed by Paul Nicholson (G8LMD), which outputs clean sine signals to a sound card.

Furthermore there is a program "*ebnaut*" from the same author, which generates the bit sequence necessary for EbNaut.

<http://abelian.org/ebnaut/ebsynth.shtml>

http://abelian.org/ebnaut/software_linux.shtml

- GPS stabilization

As already mentioned for signal reception, the transmit signal must also be highly stable in frequency, and for EbNaut also in phase. Here, too, GPS modules with PPS pulse or GPSDO oscillators have proven their worth, although the latter cannot guarantee "absolute" phase. The GPS stabilized frequency is either fed to an external modulator circuit or the PPS pulse is fed to an analog input of the sound card. There are also sound cards whose sampling rate can be stabilized by a "word clock" using GPSDO.

- Modulator circuits

As already mentioned, the modulation for various types of transmission can be controlled via the sound card. The sampling rate of the sound card has to be stabilized.

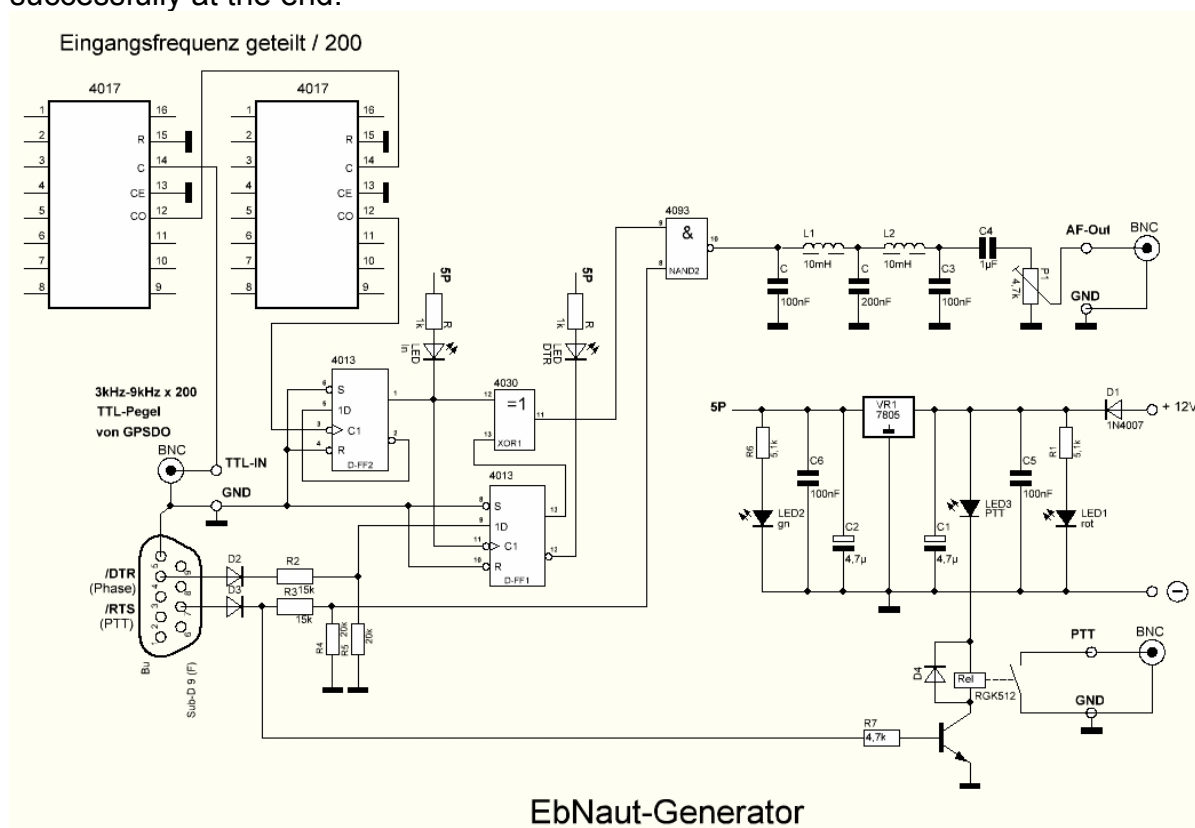
In addition, transmit modes such as CW or QRSS could also be switched via transistor switches or re- lais. The phase switching at EbNaut was realized in the beginning with relays and phase shifting. More elegantly this works now with "*ebsynth*".

Here I present my original EbNaut machine, powered by a GPSDO and controlled by the Windows program "*ebnaut-tx.exe*" via USB=>RS232 adapter.

http://abelian.org/ebnaut/software_ms.shtml

This EbNaut machine was very useful for me to make first attempts to get to know EbNaut and its pitfalls.

For this purpose the output was connected to a small antenna (for example a ferrite rod with coil) and with another antenna the signal was received weakly and fed to the sound card for the EbNaut receiving system (for example with Windows "Spectrum Lab"). So I could record the received signals on hard disk, read them out again, filter them etc. and decode them successfully at the end.



- Send with ublox GPS module

Markus Vester, DF6NM has developed a relatively simple method of using an inexpensive ublox GPS module to generate both carrier and DFCW and EbNaut signals on the desired frequency. Here is his description:

Newer GPS modules from ublox (LEA6T, NEO7, NEO8) have one or two configurable clock outputs. Usually, one output is configured for 1pps by default and drives a blinking LED. With the tool "u-center" available from the manufacturer, a connection to the GPS can be established either directly via USB or via a serial interface. Then the frequency and duty cycle of the output can be freely configured in 1 Hz steps up to over 10 MHz (menu items: u-center - View - Messages View - UBX - CFG - TP5). The output frequency is highly accurate and phase-locked to the GPS time, and is therefore very well suited for direct control of a VLF or LF transmitter. In particular (in contrast to a GPSDO with a PLL, or when using frequency dividers) the original phase is restored even after temporary failures of the satellite reception or after a restart of the system.

The square wave signal can be used directly to drive a simple class-D switching amplifier. To suppress the harmonics, either a resonant antenna circuit or (in the case of a wideband ground antenna) a dedicated power lowpass filter would be required. Alternatively, a small-signal lowpass can be used before a linear AF power amplifier. The circuit example shown here with a stepped pi element provides sufficient harmonic rejection in the 5 to 9 kHz range. Incidentally, the inductance (8.7 mH) was realized by connecting two 3.3 mH chokes side by side in series. This has two advantages: Firstly, the opposite polarity of the coils compensates for the effects of stray fields from the antenna circuit, so that no shielding is necessary. Secondly, the inductance of the coil pair is increased by approx. 30% due to the close coupling.

For the high resolution spectrograms common for VLF, several stations can transmit with very small frequency intervals (a few mHz), so that the restriction to integer 1 Hz multiples would be a severe limitation. However, it is possible to let the phase increase or decrease linearly in small steps, which allows to shift the frequency by a well-defined fraction.

For this purpose the tool "setubx2" was created for control:

<https://groups.io/g/VLF/topic/81552335>

http://df6nm.bplaced.net/LF/u-blox_ebnaut/setubx2.zip

This way a just valid phase is calculated depending on the refracted frequency part and the current system time and programmed into the GPS as "user delay". If for example "setubx2 8270.0025 >com3" is called once per second in a batch loop, the phase is advanced in small steps ($.0025 \times 360^\circ = 0.9^\circ$) until it has rotated once by 360° after 400 seconds.

Furthermore, by entering a negative frequency the output can be inverted according to the rule $\sin(-x) = -\sin(x)$. This allows a **generation of the PSK modulation for EbNaut** without an additional hardware modulator. For this there is a batch file "send_ebnaut_ublox.bat". At its beginning some variable assignments have to be edited in advance (with right click - edit):

@set raster=3600 is the start time for the program, here e.g. at the next full hour. The local time setting of the PC is valid (e.g. with raster=7200 the sequence would start at an even CET hour). The time of the PC should have been set sufficiently accurately (e.g. with NTP). The control commands are transmitted to the GPS shortly before the due date and take effect exactly at the next second start.

@set maxlate=120 means that you can "enter" a sequence up to two minutes late, e.g. because you missed the start time.

@set repeats=2 is the number of (e.g. hourly) repeats

@set sym=10 is the symbol duration in seconds (minimum 1) @set

freq=8270.0025 is the exact frequency in Hz

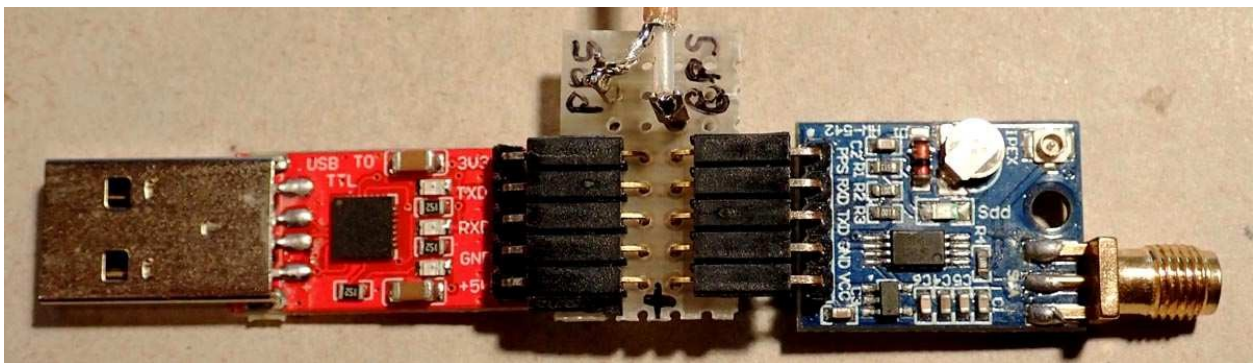
@set comport=com3 the (virtual) comport for the GPS (e.g. found in advance with u-center)

@set timepulse=1 (0 for the first or only clock output, 1 for the second clock output for a LEA6T module).

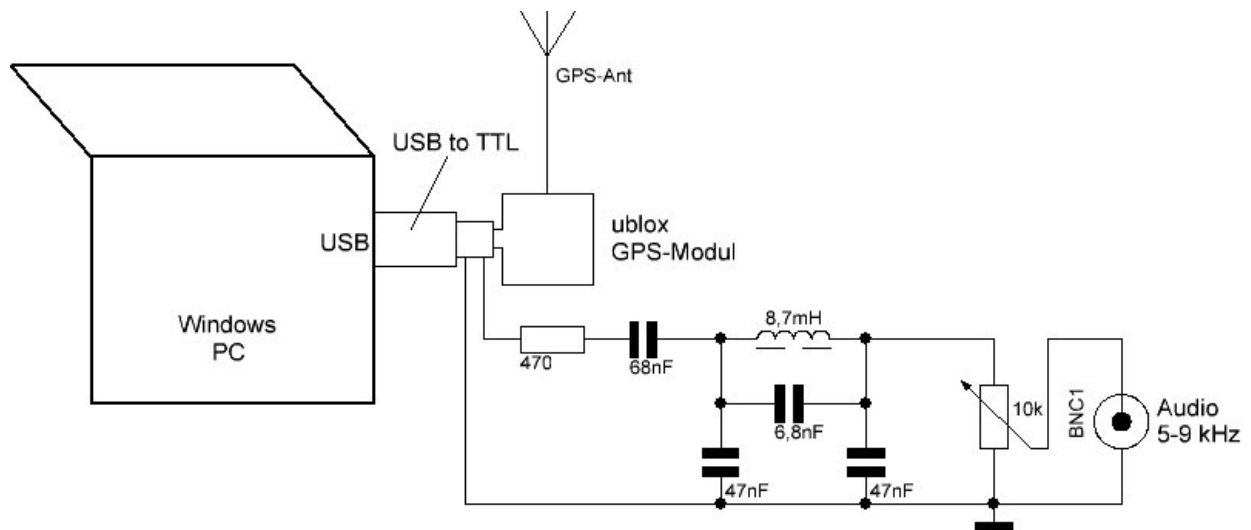
@set pskInfile=pskIn.txt This file contains a list of symbols to be sent (0 or 1, one symbol per line).

The symbol list is generated before the transmission with "ebnaut-tx.exe" (EbNaut Sender V0.9b) from Paul Nicholson. There you only have to select the encoding and enter the message itself (Message), and then save it with "Save Rows" as a text file (e.g. pskIn.txt). Further definitions of serial port and send times are not necessary for this. Afterwards the batch script can be started with a double click.

The original PPS output is branched off and the square wave signal is converted into a sinusoidal signal using a low-pass filter. This minimizes unwanted harmonics. After the TP filter, the audio signal can best be fed to the audio power amplifier via a potentiometer.



USB to serial (TTL) with adapter for branching PPS and ublox module



Overall arrangement with low-pass filter described above

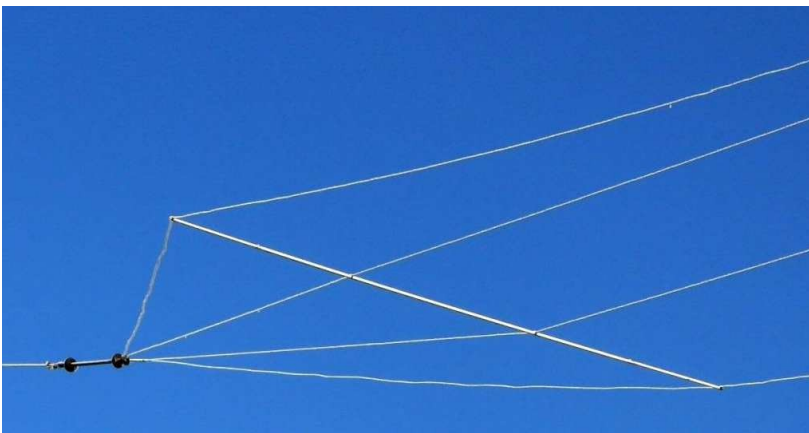
Transmitting antennas

Due to the very small transmitting antenna compared to the wavelength, its efficiency will always be poor. From 100 W of injected transmitting power, only a few μW / mW are radiated, depending on the antenna.

Because of the proximity to the conducting earth, only the vertical E-field component and the two horizontal H-field components are generally usable at very low frequencies.

- Marconi antennas / garden antennas (E-field)

Marconi antennas consist of a vertical (radiating) element and a roof capacitance, usually formed by several wires. The larger the roof capacitance and the longer (higher) the vertical radiator, the better the antenna will perform.



Roof capacity of a Marconi antenna

The matching is done by achieving series resonance with a charging coil. The charging coil should achieve a high quality with low losses. This often requires considerable large structures. (bottom right: DL3JMM at a coil for DL7NN)



Antenna loading coil is 5030 feet of No. 22 enamel magnet wire wound on a plastic barrel which once contained silicon masonry sealer. Barrel diameter is 540 mm diameter by 840 mm length. Inductance measured about 265 mH. Without the bucket variometer on top the coil tunes the antenna to 9 kHz. Adding the ageing bucket variometer provides tuning down to about 8950 Hz.



A variometer is usually placed in the large coil in order to adjust the resonance precisely. This coil is usually adapted to the audio amplifier via a transformer. A major disadvantage is the high voltage at the "hot" end of the coil. Several 10,000 V can be reached. Corona effects and flashovers can be the result.

A high hanging antenna achieves relatively good radiation values.

- Kite or balloon

If greater heights are sought, appropriate permits must be obtained as air traffic may be impacted.

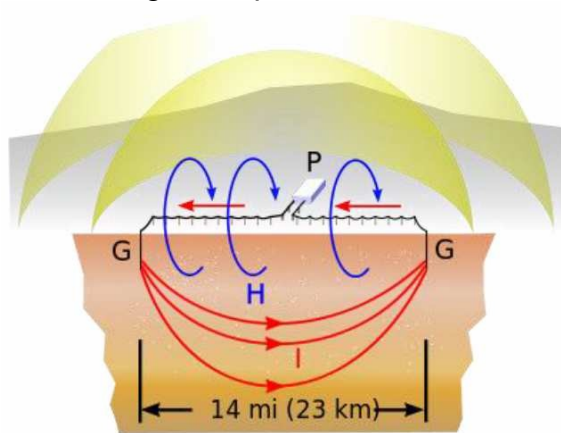
A long wire is pulled up into the air with a kite or balloon. The adjustment is done as with the Marconi antenna. Due to the movements in the wind, the resonance could change and thus the radiated phase. Here the use of "ebsynth", because in this program a phase locked loop is implemented, which is realized via a feedback of the antenna current signal.



Stefan Schäfer, DK7FC/p, mit Drachen

- Earth loops (H-field)

The ground loop is an inductive loop formed by an overhead conductor that is well grounded at both ends. The transmitter is located anywhere on the overhead conductor. The ground serves as a return conductor for the loop. The penetration depth and thus the loop area become larger the poorer the current conductivity of the underground is.



On the left, the schematic structure of an earth loop. Source:

<https://de.wikipedia.org/wiki/Bodendipol>

A suitable location has preferably rocky ground. It can be advantageous to pull the ladder over a ridge.

Lengths for amateurs are 300m...1300m.

Good grounding can be achieved, for example, on guardrails that are routed along roads.

Earth loops operate relatively broadband in frequency, one-time matching is done with a series capacitor in series with the transmitter by selecting the capacitance for the largest antenna current.

Another advantage is the low resonance resistance, which allows higher currents to flow even at relatively low voltages.

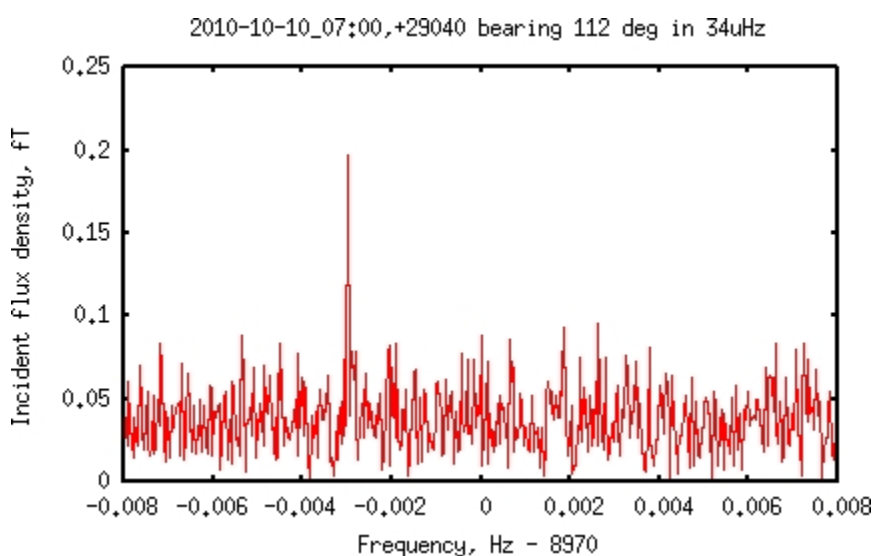
5. Send types

- Carrier shipment

The transmission of an unmodulated carrier represents both the simplest kind of a transmission and the possibility with the reception by temporal integration with narrowest bandwidth to obtain the best readability at all. As a rule, a carrier transmission is first used to test whether it makes sense to send a message, for example with EbNaut.

For the reception with Windows again the program "Spectrum Lab" is offered, with which apart from a recording on hard disk also a waterfall Spectrogram can be produced, which represents the carrier as a straight line over the time.

With Linux, the carrier can be received, filtered and recorded on disk using the "vlfri-tools". For evaluation the tool "*ebnaut*" or "*vtmspec*" can be used.



A 5 μ W ERP rubidium controlled carrier from DF6NM received at 1030km range.

<http://abelian.org/vlfrx-tools/notes.html#vtmspec> <http://abelian.org/vlfrx-tools/notes.html#Weak%20signal%20detection>

- CW, QRSS, DFCW with visually evaluated Morse code

These modes of transmission, in which the unmodulated carrier is switched or keyed in the keying pattern and slightly offset in frequency (dashes up, dots down), are no longer common on frequencies below 8.3 kHz. They can only be read in the waterfall spectrogram and are ineffective in relation to EbNaut.



- WSPR

A Windows program where different audio frequencies are digitally generated and encoded with call signs and location locator. Due to the frequency bandwidth it is not effective on **these** low frequencies.

- EbNaut

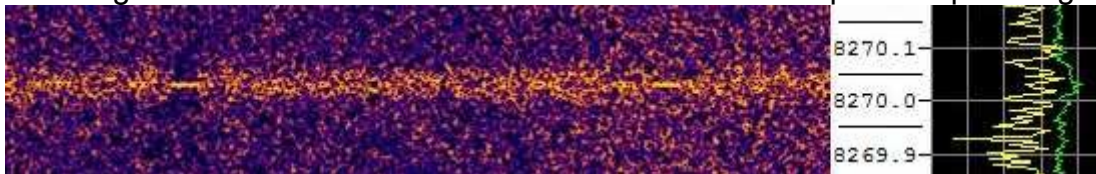
Ebnaut protocol and software was developed in 2014 by Paul Nicholson to transmit short texts for VLF. The modulation is performed as coherent phase shift keying (180°).

The decoding requires a relatively high computational effort. EbNaut does without any preambles and synchronization ("all energy for the information"). However, this requires an exact match of frequency and time.

The decoding threshold is close to the Shannon limit (E_b/N_0 -1.59 dB)

"Stacking" is possible: recording of multiple transmissions and computational overlay. This requires a reproducible absolute phase and correct times at transmitter and receiver.

EbNaut can be seen in the waterfall spectrogram only in the near field of the transmitter, weak signals do not stand out from the noise because of spectral spreading.



There are several ways to receive EbNaut:

A: Windows -> Windows (classic)

1. Recording with SpecLab and FFT data export
=> text file with 4 numbers header and complex FFT bins
2. `ebnaut_ifft3b` -> IQ wav file with inf1 header
3. `ebnaut-rx_v09` for decoding

<https://drive.google.com/file/d/1kpuFi9yDsXygNLNhnP3yAXOnvv-GYtCR/view>

B: Windows -> Windows (more elegant)

1. Time-controlled recording with SpecLab as decimated wav file
=> IQ wav file with inf1 header
2. `ebnaut-rx_v09` for decoding

https://www.qsl.net/dl4yhf/specLab/EbNaut_Rcvr.htm

C: Linux -> Linux

1. Hard disk recording on RasPi
(undecimated 48 ks/s, three channels, time correction with 1pps and vtime)
2. Extraction with command line on Raspi
> text file with three columns, e.g.

```
vtread -T2021-08-08_06:59,+5720 /raw |vtcat -p|vtmix -c1,-1,1|vtfiler -a th=5 -h  
bp,f=5170,w=2000|vtblank -a18 -d0|vtcat -S60|vtmult -f5170|vtresample -r240|vtresample -  
r1|vtraw -oa >/raw/extracted/NE_5k17_210808_0700.txt
```

3. ebnaut correlation with command line e.g.

```
ebnaut </raw/extracted/NW_5k17_210808_0700.txt -dp8k19a -N6 -r1 -S10 -F.03 -f16 -f15 -v  
-M'JO60MR'
```

Correlation with a known message is useful to detect weak EbNaut signals below the decoder threshold.

D: Linux -> Windows

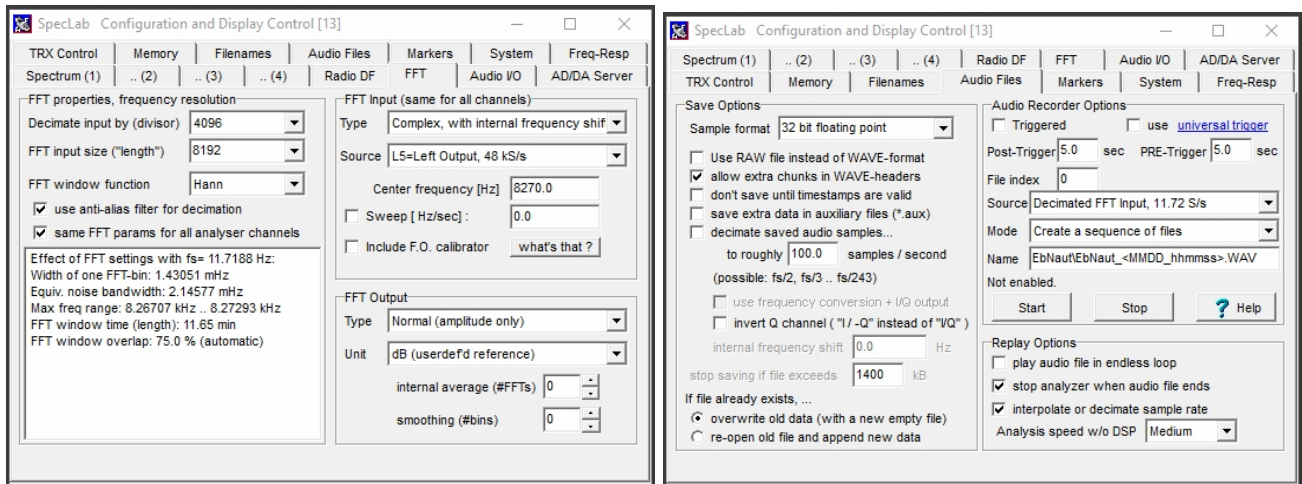
1. and 2. as above for method C
3. Copy txt file to Windows
4. `vttowav.exe` -> IQ wav file with inf1 headers
5. `ebnaut-rx_v09.exe` decode under Windows

E: Windows -Linux

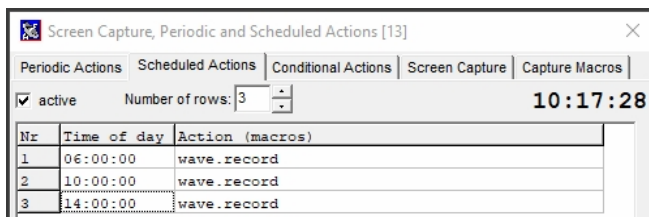
1. Recording with SpecLab and decimated wav export
=> IQ wav file with inf1 header
2. `wavtovt` -> three-column txt with IQ data
3. Copy txt file to Linux
4. Decode or correlate with ebnaut under Linux, as with C 3.

For now, we will only describe variants B and D in detail:

Variant B: Windows => Windows



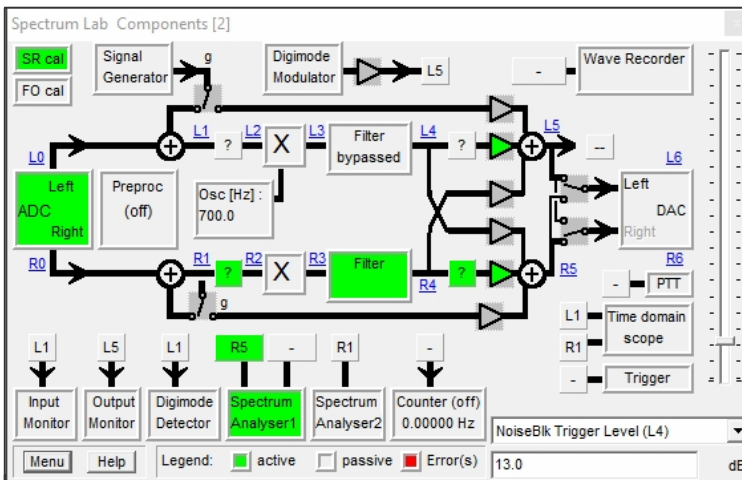
In the Windows program Spectrum Lab set the options "FFT" and "Audio Files" accordingly.



Call up the "Scheduled Actions" card under "File" and set the times for the EbNaut transfers.

The first transmission starts at 06:00 UTC, at this time the recording of a WAV file is started and at 10:00 UTC the recording of a WAV file is started.

UTC is stopped and the next WAV file is started.



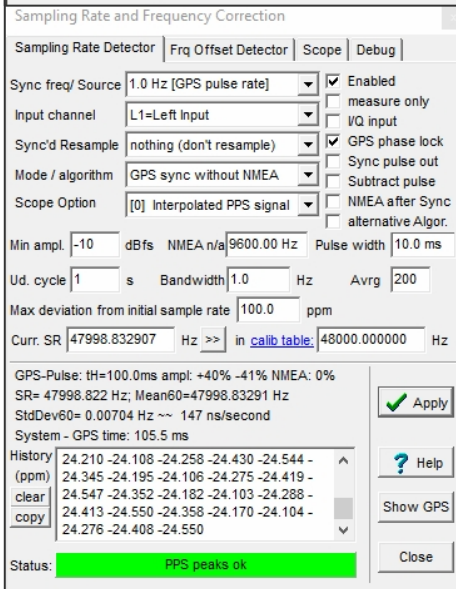
In "Components" filters and noise blankers are set for the RX frequency.

Click on the "SR cal" field in "Components" and activate the correction of the sampling rate by the PPS clock.

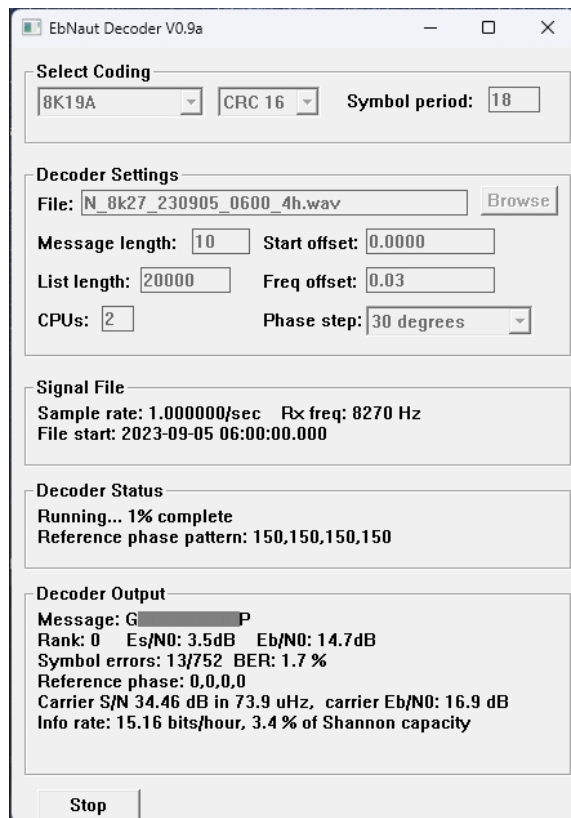
If everything works, "Status" turns green.

Under Scope the "Min ampli" threshold can be controlled.

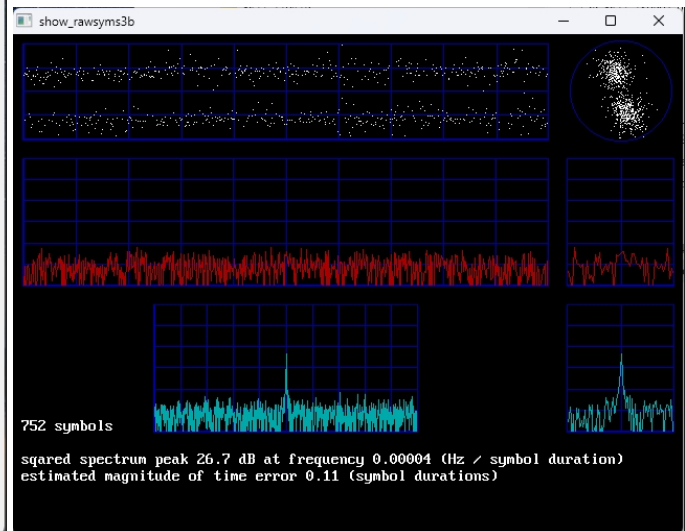
The time must also be set exactly to UTC. This can be realized via the Internet or also from the GPS module via USB.



The created WAV file is then decoded with "*ebnaut-rx.exe*":



http://abelian.org/ebnaut/software_ms.shtml



If exactly 8270 Hz was selected as "center frequency" in the SpecLab, but the EbNaut transmitter transmitted at 8270.03 Hz, the low frequency offset can be set here.

If everything has worked, the correct decoding with the data of the transmission appears after some time. A protocol and the file "*rawsyms.txt*" are written into the directory of the WAV file. The *rawsym.txt* can be opened with the mouse on the file "*show_rawsyms3b*" (from DF6NM), then the above right picture is created. At the top of the picture you can see the course of the phase switches. There must be 2 straight lines. Below are hints about frequency and time error.

http://df6nm.bplaced.net/VLF/fec_tests/df6nm_ebnaut_utilities.zip

Log file example:

```
ebnaut-rx V0.9a
input file D:\Spectrum\EbNaut_1029_110000.WAV sample rate
46.875000 per second
rx frequency 227.000000000
file start time 2022-10-29 10:59:47.864
polynomial 16K21A
crc size 16
number of chars 5
block size 46
symbol period 14.000
number of symbols 1056
start offset 12.0000
freq offset 0.004400
cores 1
list length 20000
skipped 12.010667 seconds to start
padded 4692.203 seconds at end
initial reference phase -5.1
```

If the start time for the WAV file in this decoding protocol file is different from the start time (here 11:00 UTC), this difference (here 12.14s) must be entered in **the decoder** at "Start offset".

Variant D: Linux => Windows

Prerequisites:

It is assumed here that only one antenna leads to channel 2 of the sound card. The vlfrx-tools and *ebnaut* are installed (on the Raspi), into channel 1 (left) of the soundcard the PPS signal is led, the system time is more accurate than 0.5s, the signals of the soundcard are recorded with *vtcard* and *vttime* works. The buffer of *vtcard* was called *@raw*.

Stream files and named pipes use normal Unix path names. Unlocked pipes use a buffer name prefixed with an '@' to distinguish it from a pathname.

In file *etc/rc.local* the path and the file is entered: */home/pi/vt_start.sh*

In *vt_start.sh* all programs are called that retrieve and process signals from the sound card. The program must be made executable:

```
sudo chmod +x vt_start.sh
```

In */dev/shm* you can see the temporary files which are *the* source of the VLF tools with *@xxx*.

Here you can see my start file, it has to be modified according to the sound card and the other conditions and needs:

```
#!/bin/bash
```

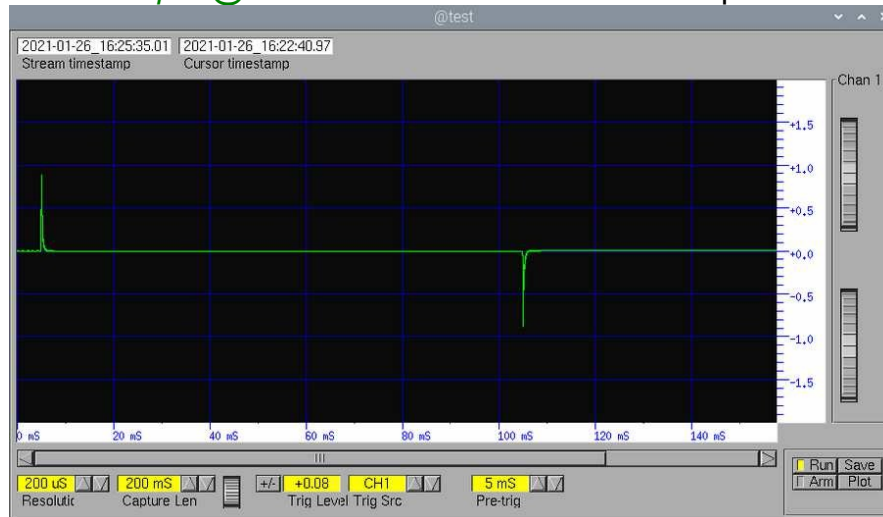
```
sudo ifconfig eth0 169.254.171.100  
sudo vtcard -B -r48000 -c20 -b32 -d usb:1-1.1 @raw,i2  
vtwait -t @raw  
vttime -B -m edge+ @raw:1,2,3,4 @timed  
vtwait -t @timed  
vtcat -B -p @timed:2,3 @res,i2  
sudo vtwrite -G43200 @res  
/mnt/roh
```

Especially with *vtcard* the parameters must be adapted to the used sound card, otherwise there will be problems with the data connection. With *-cxx* the number of channels of the sound card is specified (-c20 for Tascam 208i). It is also possible to change the default setting for the buffers. In case of doubt it can be helpful to read the description for the software.

<http://abelian.org/vlfrx-tools/notes.html#vtcard>

<http://abelian.org/vlfrx-tools/notes.html#Soundcards>

Now *vtscope: @raw:1 can* be called. An oscilloscope is created:



The PPS signal on channel 1 should look like this or similar.

With *vtstat @raw* and *vtstat @timed* the function can be controlled.

Next, the audio stream from *vttime* is written to disk: Either the original stream

sampled at 48 kHz is recorded, or with

vtresample -r 24000 is reduced to 24 kHz.

```
sudo vtresample -r24000 -B @timed:2 @res
sudo vtwrite -B -G86400 @res /home/pi
```

With these commands the audio stream is first reduced in the sampling rate to 24 kHz, so that frequencies up to approx. 9 kHz can be evaluated. In the above example, *vtwrite* is used to write to the **SD card**. If several days are to be recorded, the file path of the **external hard disk** should be specified as the destination. Mostly the HD has to be mounted before. The data is stored one after the other in files, usually with a day length, and can be read out across files with *vtread*.

In case of doubt, it is recommended to refer to the original software description:

<http://abelian.org/vlfrx-tools/>

If an EbNaut message has been sent and the data is known, the following commands can be used to read out the period of the transmission, filter it, mix it down to baseband, reduce it to 1 - 2 samples/second and generate a text file.

To allow the notch filter and the noise blanker to settle in, readout is started 60s beforehand, and the 60s are removed again after the filter with *vtcat -S60*.

Example:

<code>vtread -T2021-08-08_06:59,+5720 /raw</code>	Read, time, length (s)
<code> vcat -p</code>	Zeros are filled
<code> vfilter -a th=5 -h bp,f=5170,w=2000</code>	Hum and bandpass filter 5.17 kHz / 2 kHz
<code> vtblank -a18 -d0</code>	Noiseblanker
<code> vcat -S60</code>	-60s
<code> vtmult -f5170</code>	mix down to baseband
<code> vtresample -r240</code>	Reduce sample rate to 240 S/s
<code> vtresample -r1</code>	Reduce sample rate to 1 S/s
<code> vtraw -oa >/raw/extracted/NE_5k17_210808_0700.txt</code>	Create text file in directory

The commands are written one after the other, the | character ensures the data transfer:

```
vtread -T2021-08-08_06:59,+5720 /raw |vcat -p| vfilter -a th=5 -h bp,f=5170,w=2000|vtblank -a18 -d0|vcat -S60|vtmult -f5170|vtresample -r240|vtresample -r1|vtraw -oa >/raw/extracted/NE_5k17_210808_0700.txt
```

The resulting text file is copied to Windows and converted to a WAV file with "vttowav.exe" from DF6NM. (txt, wav, bat and "vttowav.exe" in the same directory).

http://df6nm.bplaced.net/VLF/fec_tests/vttowav.zip

It is best to create a batch file "vttowav_5170.bat" for the selected center frequency with only one line:

```
>vttowav %1 5170 <
```

Then drag and drop the icon of the txt file onto the .bat icon. The conversion program will run for a few seconds and the WAV file is ready.

Decoding with **ebnaut-rx.exe**

From here on the same decoding actions are executed as in variant B. The number of processor cores can be specified and shortens the computing time.

There are several other ways to decode EbNaut, for example, it can be decoded directly in Linux using the "ebnaut" command. Especially with strong codes (16K23, 16K25) or longer messages, the decoder needs a considerable amount of computational effort and a lot of RAM memory. An average equipped Raspberry Pi is often not sufficient.

However, "ebnaut" can calculate the correlation to a **known message with** the parameter -f16. Less resources are used and the calculation takes less time.

Example:

```
ebnaut </raw/extracted/NW_5k17_210808_0700.txt -dp8k19a -N6 -r1 -S10 -F.03 -f16 -f15 -v -M'JO60MR'
```

Stacking WAV files

If the EbNaut message was sent with several repetitions, a further improvement of the SNR can be achieved by stacking the WAV files.

For this purpose the tool "*addwav1.exe*" from DF6NM is used.

If you drag the first file over the icon of *addwav1.exe*, it will be converted to a new "sum.wav" is copied. Each additional WAV file is then added to "sum.wav". If it is called with the command line, individual weight factors and time corrections can also be specified

http://df6nm.bplaced.net/VLF/fec_tests/addwav1.zip

For stacking, Paul Nicholson has also written a program that would be worth testing.

Link to *./stack2*:

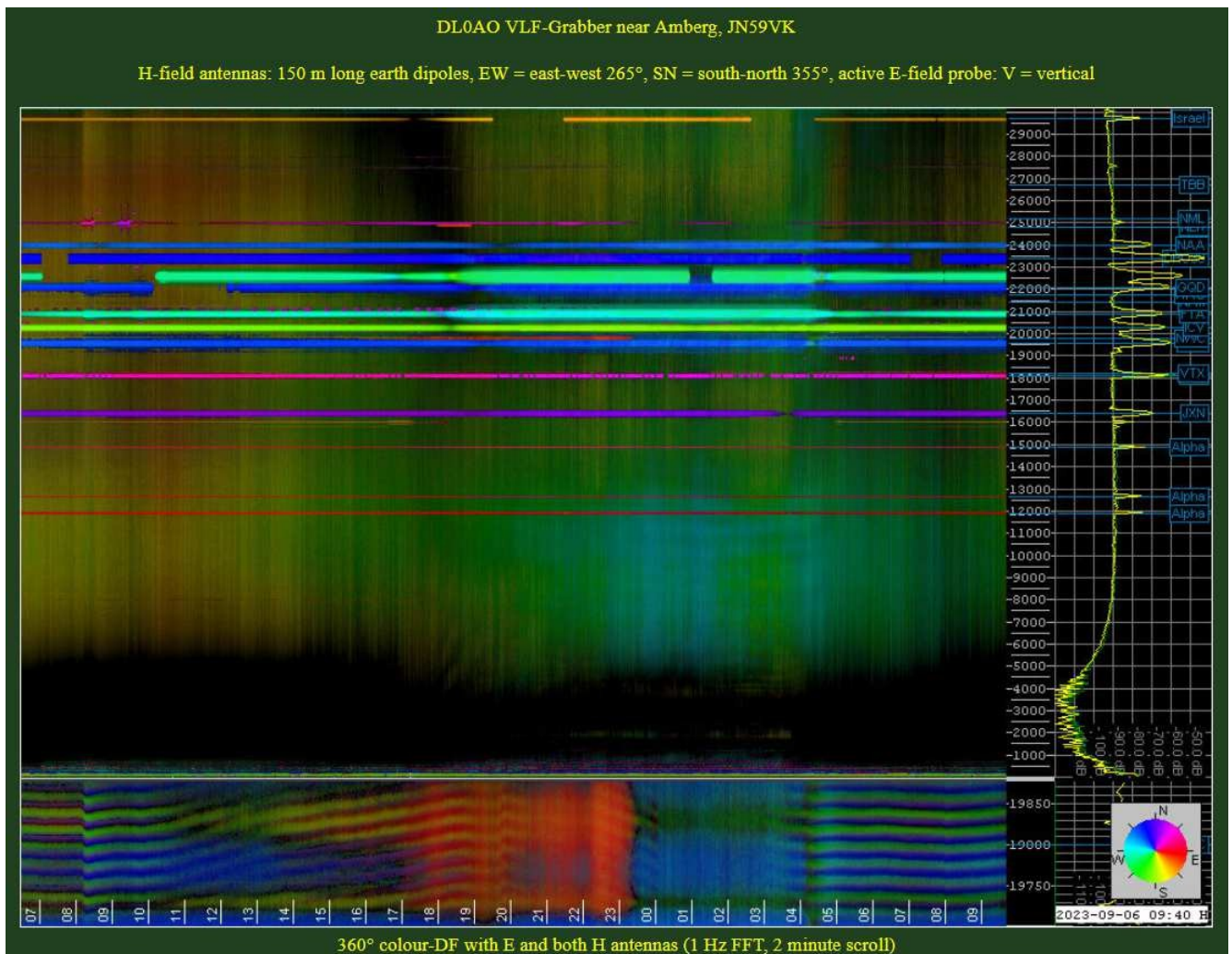
<https://groups.io/g/VLF/message/29678>

6. Grabber

Grabbers are automated receiving stations, usually located in low-interference environments, that provide their reception results as waterfall spectrograms over the Internet.

Some operators also have a Linux computer running in parallel, on which they record the audio streams on hard disk.

- normal grabber



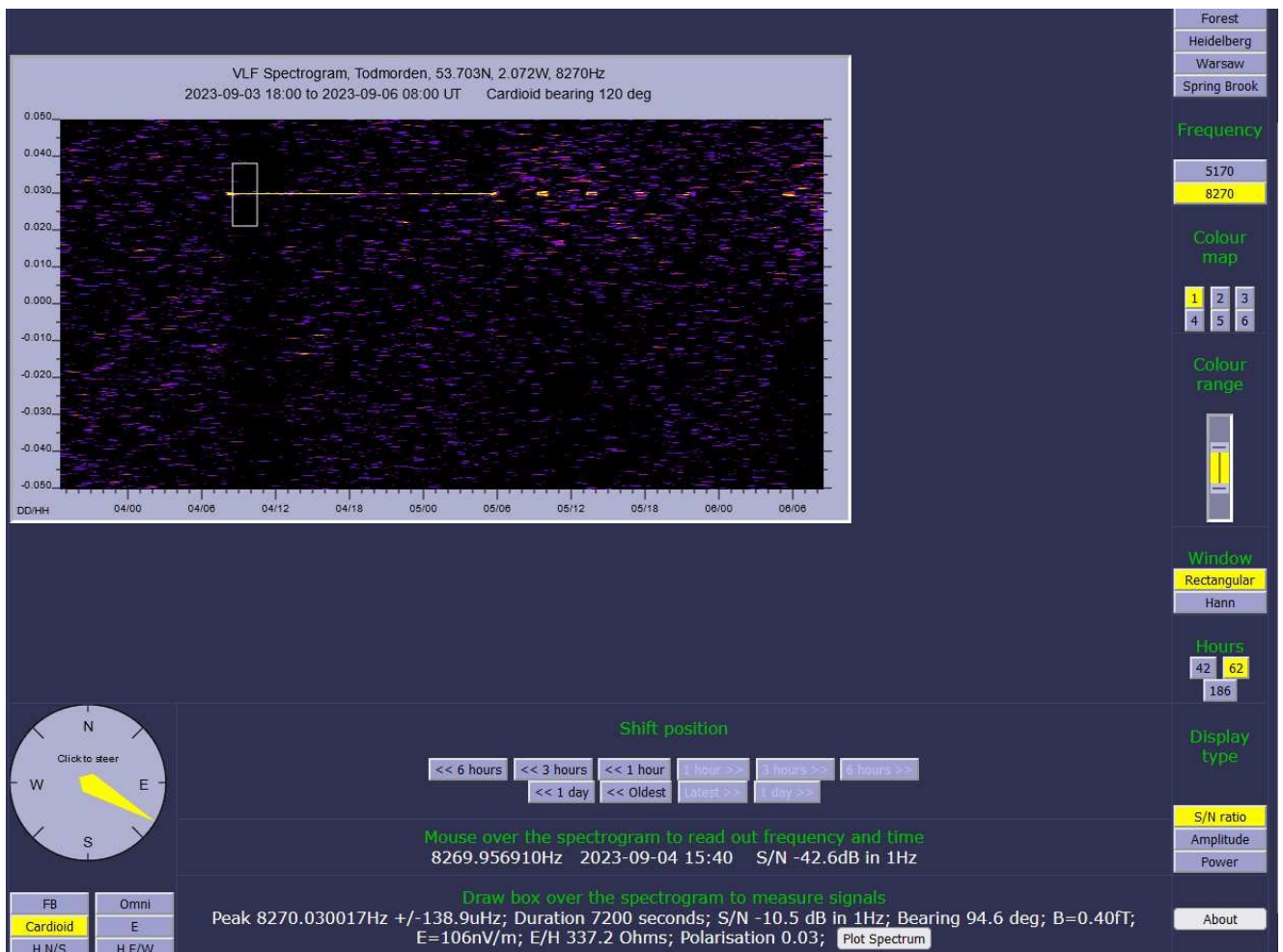
Here DL0AO, he shows many pictures with different frequencies and antenna signals

<https://vlf.u01.de/>

- Paul's configurable and switchable grabber

Various antennas and directional diagrams, as well as audio streams of some other stations can be set and viewed here. Furthermore, the field strength of adjustable screen sections is displayed with the corresponding selection and with "Plot Spectrum" a high-resolution spectrum can be created from it.

<http://5.9.106.210/fbins3.html#r=vlf1&f=8270&p=1693987200&b=120&s=sp&m=cardioi&d&w=r&h=62&z1=0.34&z2=0.64&c=1>

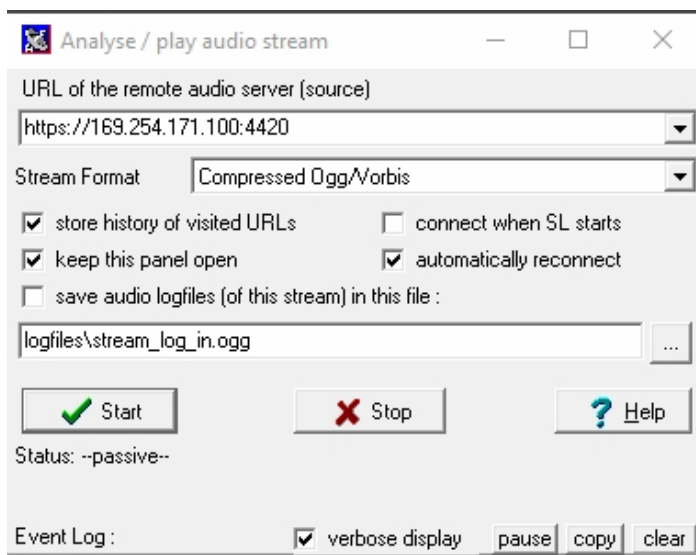


- Receive streams from different stations

This website by Paul Nicholson gives web addresses where you can live stream audio from various stations via "Vorbis" and receive, filter, record and decode it with an independent system.

<http://abelian.org/vlf/index.php?page=access>

Under SpecLab: *File / Audio File & Stream analysis / Analyze and play stream/URL:*



Enter the corresponding URL here and press Start, the stream will be read in as input for further processing.

- Some links of currently active grabbers:

DF6NM (Nuremberg):

<http://www.df6nm.de/vlf/vlfgrabber.htm>

Paul Nicholson (Todmorden, UK):

<http://5.9.106.210/fbins3.html#r=vlf1&f=8270&p=1693987200&b=000&s=sp&m=cardioi&d&w=r&h=62&z1=0.34&z2=0.64&c=1>

DL0AO (Amberg):

<https://vlf.u01.de/>

DK7FC (Heidelberg):

http://web1.iup.uni-heidelberg.de/schaefer_vlf/DK7FC_VLF_Grabber2.html

KL7L (Wasilla, Alaska):

<https://kl7l.com/>

Renato Romero (Cumiana, Northern Italy):

<http://www.vlf.it/cumiana/livedata.html>

7. Closing words

This description should serve as an introduction for beginners, in this form it is not possible to explain comprehensively all the paths and side paths taken so far.

I recommend to study the detailed descriptions of the software to be able to use its extensive functionality.

For me it was very helpful to test with a weak transmitter from one room to another, receive with an independent system, filter, record and decode.

Various commercial transmitters such as ZEVS (82 Hz) can serve as indicators for the receivability of the receiving systems, as well as the reception of the Schumann resonances. Often a comparison with the reception results of various grabbers available on the Internet also helps.

Many suggestions and announcements are posted in the VLF- Reflektor. Here you can learn and discuss a lot:

<https://groups.io/g/VLF/topics>

Source for some of the images used:

http://df6nm.bplaced.net/VLF/Afu_Tagung/